



درس طراحی زبان‌های برنامه‌سازی

دکتر محمد ایزدی

تمرین اول

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال دوم ۱۴۰۰ - ۱۳۹۹

مهلت ارسال:

۲۰ فروردین ۱۴۰۰

ساعت ۲۳:۵۹



به موارد زیر توجه کنید:

- * برنامه‌های خود را به زبان Racket بنویسید.
- * مهلت ارسال تمرین ساعت ۲۳:۵۹ روز ۲۰ فروردین ۱۴۰۰ است.
- * در مسائلی که نیاز به خروجی دادن است، پاسخ را چاپ نکنید بلکه آن را به عنوان حاصل تابع برگردانید.
- * حتما نام تابع اصلی خود را main بگذارید تا هنگام داوری به مشکل نخورید. منظور از تابع اصلی تابعی است که اجرای آن روی ورودی، خروجی مطلوب را نمایش می‌دهد.
- * جواب هر سوال را در یک فایل Racket با نام و فرمت زیر قرار دهید.
$$[StudentID]_[ProblemNumber].rkt$$

مثلا نام فایل مربوط به مسئله ۳ : $90123456_3.rkt$
- * در نهایت تمام فایل‌های رکت را در یک فایل زیپ قرار داده و با نام $HW1_StudentID$ در سامانه کوئرا آپلود کنید.
- * هرگونه سوالی راجع به این تمرین را در زیر پست مربوطه در کوئرای درس مطرح کنید.
- * در مجموع تمامی تمرین ۷ روز مهلت تاخیر مجاز دارید و پس از تمام شدن این تاخیرهای مجاز به ازای هر روز ۱۰ درصد از کل نمره تمرین شما کم می‌شود.
- * لطفا تمرین‌ها را از یکدیگر کپی نکنید. در صورت وقوع چنین مواردی مطابق با سیاست درس رفتار می‌شود.



۱ کار با لیست ۱

برنامه‌ای بنویسید که دو ماتریس را به عنوان ورودی گرفته و ضرب آنها را در خروجی بدهد. تضمین می‌شود اندازه‌های دو ماتریس برای عمل ضرب به مشکل نخورند.

نمونه فراخوانی

```
(main '((2 1) (4 3)) '((-1) (1)))
```

خروجی مطلوب

```
'((-1) (-1))
```

۲ کار با فایل

برنامه‌ای بنویسید که مسیر فایل ورودی و خروجی را دریافت کرده، کد رکت موجود در فایل ورودی را اجرا کند و مقدار خروجی را در مسیر خروجی ذخیره کند. در صورت وجود فایل خروجی ابتدا آن فایل را پاک کرده و سپس مجدد آن فایل را ایجاد کنید. (راهنمایی: از eval استفاده کنید.)

نمونه فراخوانی

```
(main "input.rkt" "output.txt")
```

نمونه فایل‌های ورودی و خروجی input.rkt و output.txt در فایل سوال قرار داده شده است.



۳ کار با لیست ۲

برنامه‌ای بنویسید که یک عبارت prefix را دریافت کرده و حاصل عبارت را خروجی دهد. عبارت ورودی تنها شامل $+$ $-$ $*$ $/$ و عدد خواهد بود. برای یادگیری عبارات prefix میتوانید از لینک زیر استفاده کنید.

<http://www.cs.man.ac.uk/~pjj/cs212/fix.html>

نمونه فراخوانی

```
(main '(+ * 8 6 / 20 4))
```

خروجی مطلوب

53

در مثال بالا ورودی درواقع عبارت $20/4 + 6 * 8$ است.

۴ کار با تابع و exception

برنامه‌ای بنویسید که لیستی از توابع را به عنوان ورودی دریافت کند. سپس ۱- تابع اول را بدون ورودی اجرا کند. ۲- بقیه توابع را صدا بزند و به عنوان ورودی هر تابع، خروجی حاصل از اجرای تابع قبل را بدهد. ۳- خروجی تابع آخر را برگرداند. در هر کدام از این مراحل در صورتی که ارور رخ داد رشته‌ی "error" را برگردانید. دقت کنید که در هیچ حالتی نباید اجرای کلی برنامه ارور بدهد.

نمونه فراخوانی

```
(main (list f g h))
```

خروجی مطلوب



در این مثال تابع f عدد ۵ را برمیگرداند، تابع g عدد ورودی را به توان ۲ میرساند و تابع h یکی از عدد ورودی کم می‌کند.

۵ کار با لیست ۳

تعریف درخت دودویی جستجو در صفحه ۱۰ کتاب Programming of Essentials Languages (ویرایش سوم) را در نظر بگیرید. تابعی بنویسید که یک لیست یک سطحی از اعداد را به عنوان ورودی بگیرد و یک درخت دودویی جستجو به عنوان خروجی بدهد. در الگوریتم ساخت این درخت جستجو، اعداد را به ترتیب از لیست بخوانید و در جای مناسب درخت بگذارید.

نمونه فراخوانی

```
(main '(2 1 3))
```

خروجی مطلوب

```
'(2 (1 () (3 () ())))
```